In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from IPython import get_ipython
import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```
fetal_data = pd.read_csv('fetal_health.csv')
```

In [3]:

```
fetal_data.head()
```

Out[3]:

	baseline value	accelerations	fetal_movement	uterine_contractions	light_decelerations	severe_dece
0	120	0.000	0.0	0.000	0.000	
1	132	0.006	0.0	0.006	0.003	
2	133	0.003	0.0	0.008	0.003	
3	134	0.003	0.0	0.008	0.003	
4	132	0.007	0.0	0.008	0.000	

5 rows × 22 columns

In [4]:

```
fetal_data.tail()
```

Out[4]:

	baseline value	accelerations	fetal_movement	uterine_contractions	light_decelerations	severe_c
2121	140	0.000	0.000	0.007	0.0	
2122	140	0.001	0.000	0.007	0.0	
2123	140	0.001	0.000	0.007	0.0	
2124	140	0.001	0.000	0.006	0.0	
2125	142	0.002	0.002	0.008	0.0	

5 rows × 22 columns

In [5]:

```
1 fetal_data.shape
```

Out[5]:

```
(2126, 22)
```

In [6]:

```
1 fetal_data.columns
```

Out[6]:

```
Index(['baseline value', 'accelerations', 'fetal_movement',
       'uterine_contractions', 'light_decelerations', 'severe_deceleration
s',
       'prolongued_decelerations', 'abnormal_short_term_variability',
       'mean_value_of_short_term_variability',
       'percentage_of_time_with_abnormal_long_term_variability',
       'mean_value_of_long_term_variability', 'histogram_width',
       'histogram_min', 'histogram_max', 'histogram_number_of_peaks',
       'histogram_number_of_zeroes', 'histogram_mode', 'histogram_mean',
       'histogram_median', 'histogram_variance', 'histogram_tendency',
       'fetal_health'],
      dtype='object')
```

In [7]:

```
1 fetal_data.duplicated().sum()
```

Out[7]:

```
13
```

In [8]:

```
1 fetal_data = fetal_data.drop_duplicates()
```

In [9]:

```
1 fetal_data.shape
```

Out[9]:

```
(2113, 22)
```

In [10]:

```
1 fetal_data.isnull().sum()
```

Out[10]:

```
baseline value                                            0
accelerations                                             0
fetal_movement                                            0
uterine_contractions                                      0
light_decelerations                                       0
severe_decelerations                                      0
prolongued_decelerations                                  0
abnormal_short_term_variability                           0
mean_value_of_short_term_variability                      0
percentage_of_time_with_abnormal_long_term_variability    0
mean_value_of_long_term_variability                       0
histogram_width                                           0
histogram_min                                             0
histogram_max                                             0
histogram_number_of_peaks                                 0
histogram_number_of_zeroes                                0
histogram_mode                                            0
histogram_mean                                            0
histogram_median                                          0
histogram_variance                                        0
histogram_tendency                                        0
fetal_health                                              0
dtype: int64
```

In [11]:

```
1 fetal_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2113 entries, 0 to 2125
Data columns (total 22 columns):
 #   Column                                                  Non-Null Coun
t  Dtype
---  ------                                                  -------------
-  -----
 0   baseline value                                          2113 non-null
int64
 1   accelerations                                           2113 non-null
float64
 2   fetal_movement                                          2113 non-null
float64
 3   uterine_contractions                                    2113 non-null
float64
 4   light_decelerations                                     2113 non-null
float64
 5   severe_decelerations                                    2113 non-null
float64
 6   prolongued_decelerations                                2113 non-null
float64
 7   abnormal_short_term_variability                         2113 non-null
int64
 8   mean_value_of_short_term_variability                    2113 non-null
float64
 9   percentage_of_time_with_abnormal_long_term_variability  2113 non-null
int64
 10  mean_value_of_long_term_variability                     2113 non-null
float64
 11  histogram_width                                         2113 non-null
int64
 12  histogram_min                                           2113 non-null
int64
 13  histogram_max                                           2113 non-null
int64
 14  histogram_number_of_peaks                               2113 non-null
int64
 15  histogram_number_of_zeroes                              2113 non-null
int64
 16  histogram_mode                                          2113 non-null
int64
 17  histogram_mean                                          2113 non-null
int64
 18  histogram_median                                        2113 non-null
int64
 19  histogram_variance                                      2113 non-null
int64
 20  histogram_tendency                                      2113 non-null
int64
 21  fetal_health                                            2113 non-null
int64
dtypes: float64(8), int64(14)
memory usage: 379.7 KB
```

In [12]:

```
1 fetal_data.describe()
```

Out[12]:

	baseline value	accelerations	fetal_movement	uterine_contractions	light_decelerations	sev
count	2113.000000	2113.000000	2113.000000	2113.000000	2113.000000	
mean	133.304780	0.003188	0.009517	0.004387	0.001901	
std	9.837451	0.003871	0.046804	0.002941	0.002966	
min	106.000000	0.000000	0.000000	0.000000	0.000000	
25%	126.000000	0.000000	0.000000	0.002000	0.000000	
50%	133.000000	0.002000	0.000000	0.005000	0.000000	
75%	140.000000	0.006000	0.003000	0.007000	0.003000	
max	160.000000	0.019000	0.481000	0.015000	0.015000	

8 rows × 22 columns

In [13]:

```
1 fetal_data.nunique()
```

Out[13]:

```
baseline value                                          48
accelerations                                           20
fetal_movement                                         102
uterine_contractions                                    16
light_decelerations                                     16
severe_decelerations                                     2
prolongued_decelerations                                 6
abnormal_short_term_variability                         75
mean_value_of_short_term_variability                    57
percentage_of_time_with_abnormal_long_term_variability  87
mean_value_of_long_term_variability                    249
histogram_width                                        154
histogram_min                                          109
histogram_max                                           86
histogram_number_of_peaks                               18
histogram_number_of_zeroes                               9
histogram_mode                                          88
histogram_mean                                         103
histogram_median                                        95
histogram_variance                                     133
histogram_tendency                                       3
fetal_health                                             3
dtype: int64
```

In [14]:

```
1 fetal_data['fetal_health'].unique()
```

Out[14]:

```
array([2, 1, 3], dtype=int64)
```

In [15]:

```
1 fetal_data['fetal_health'].value_counts()
```

Out[15]:

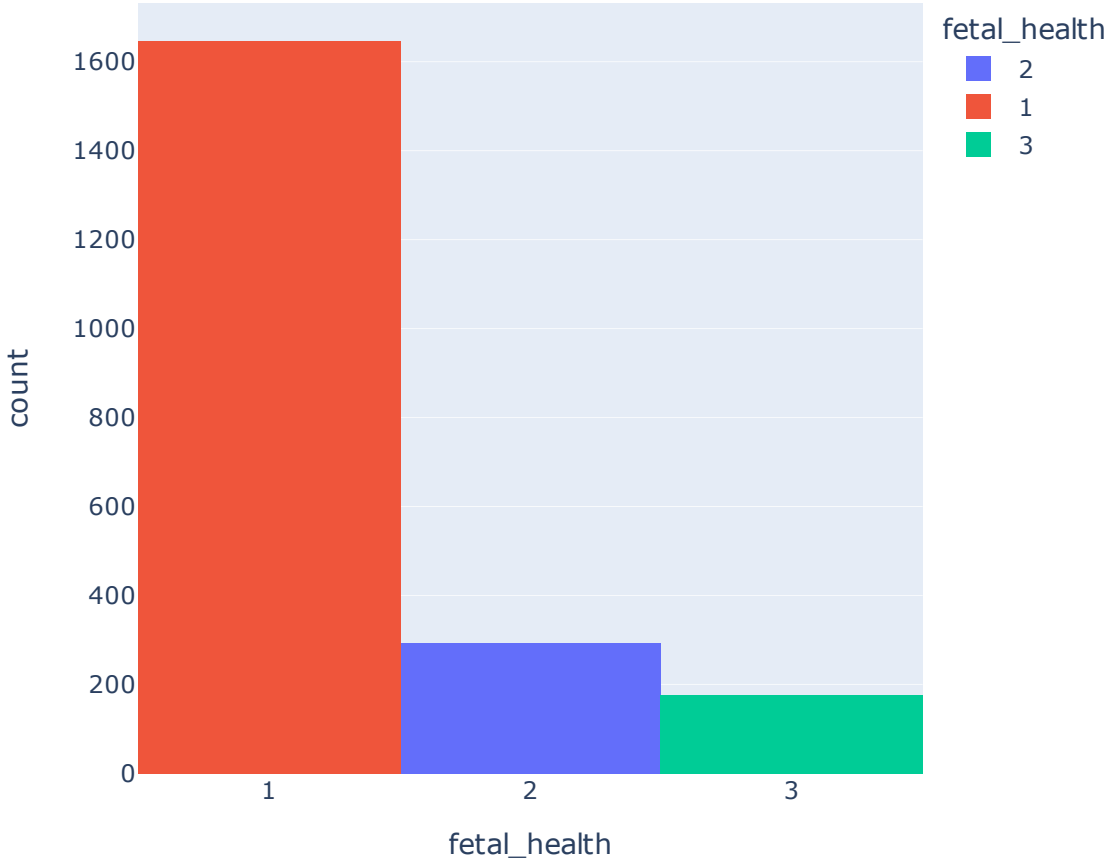
```
1    1646
2     292
3     175
Name: fetal_health, dtype: int64
```

In [16]:

```
1 import plotly.express as px
```

In [17]:

```
fig1 = px.histogram(fetal_data, x = 'fetal_health',
                    color = 'fetal_health')
fig1.show()
```

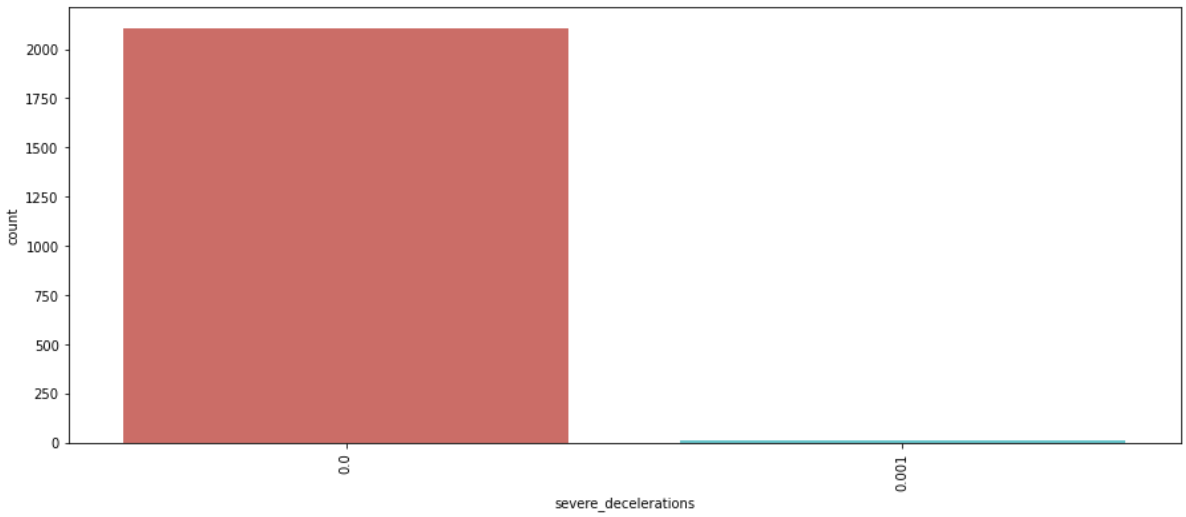


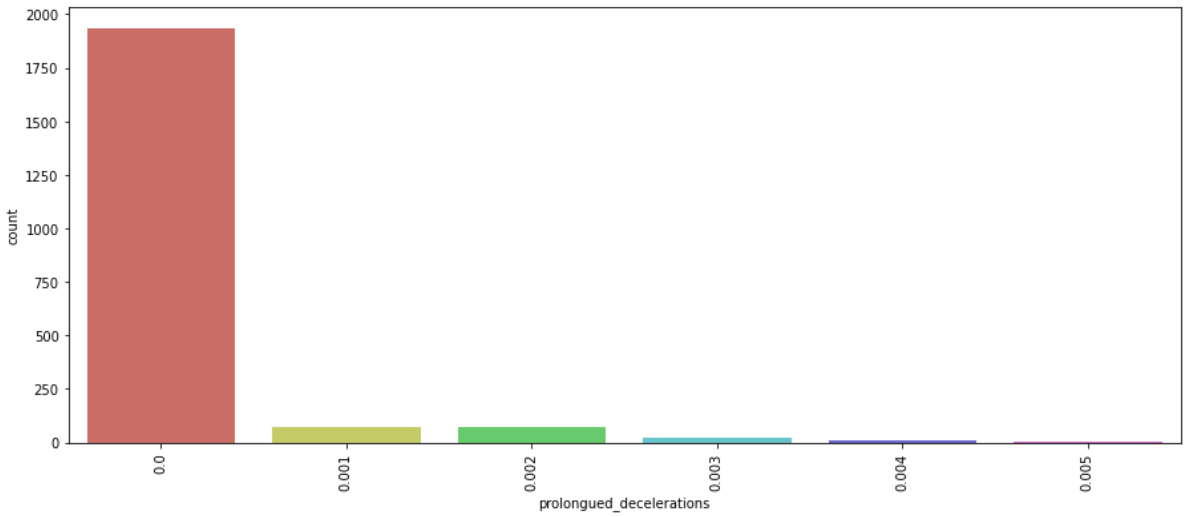
In [18]:

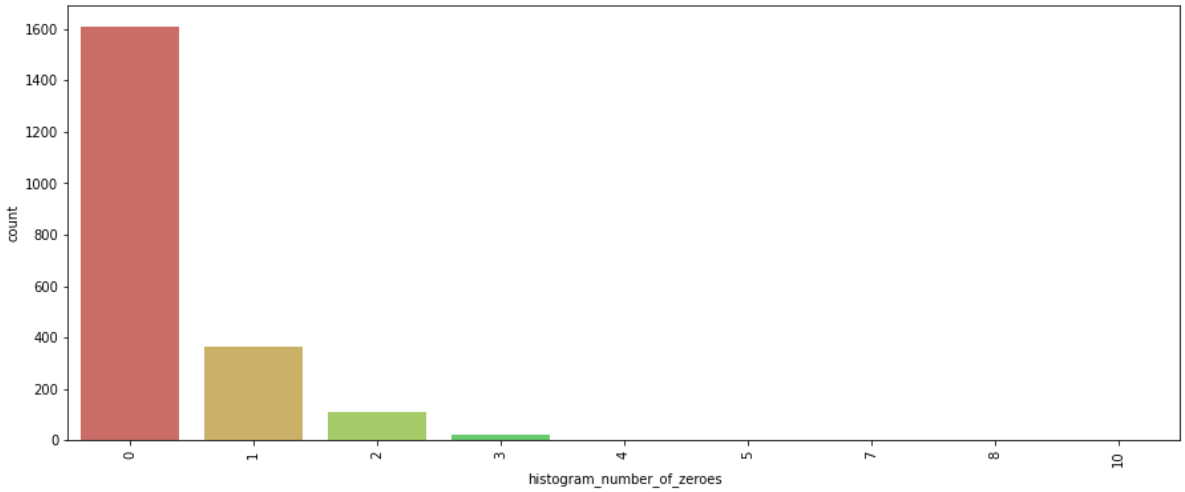
```
fetal_data_cat = fetal_data[['severe_decelerations', 'prolongued_decelerations',
                             'histogram_number_of_zeroes', 'histogram_tendency']]
```

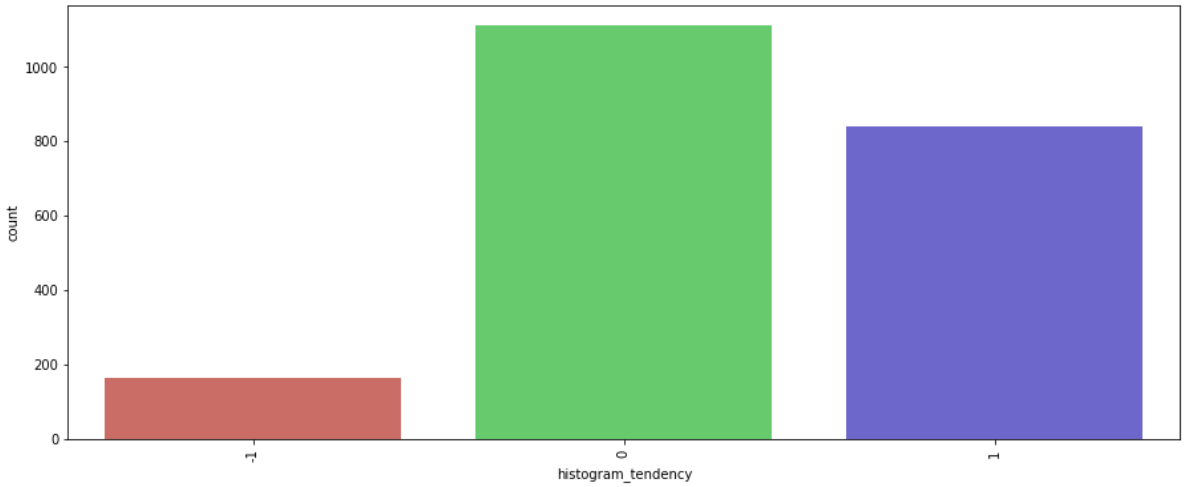
In [19]:

```
for i in fetal_data_cat.columns:
    plt.figure(figsize=(15,6))
    sns.countplot(fetal_data_cat[i], data = fetal_data_cat,
                  palette='hls')
    plt.xticks(rotation = 90)
    plt.show()
```



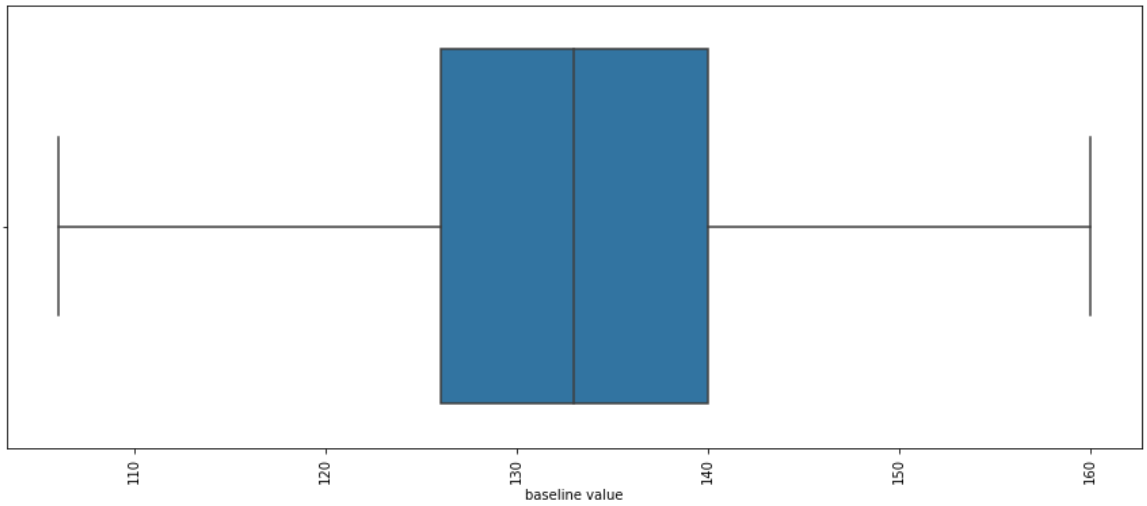



1600
1400
1200
1000
800
600
400
200
0
count
0
1
2
3
4
5
7
8
10
histogram_number_of_zeroes


1000
800
600
400
200
0
count
-1
0
1
histogram_tendency

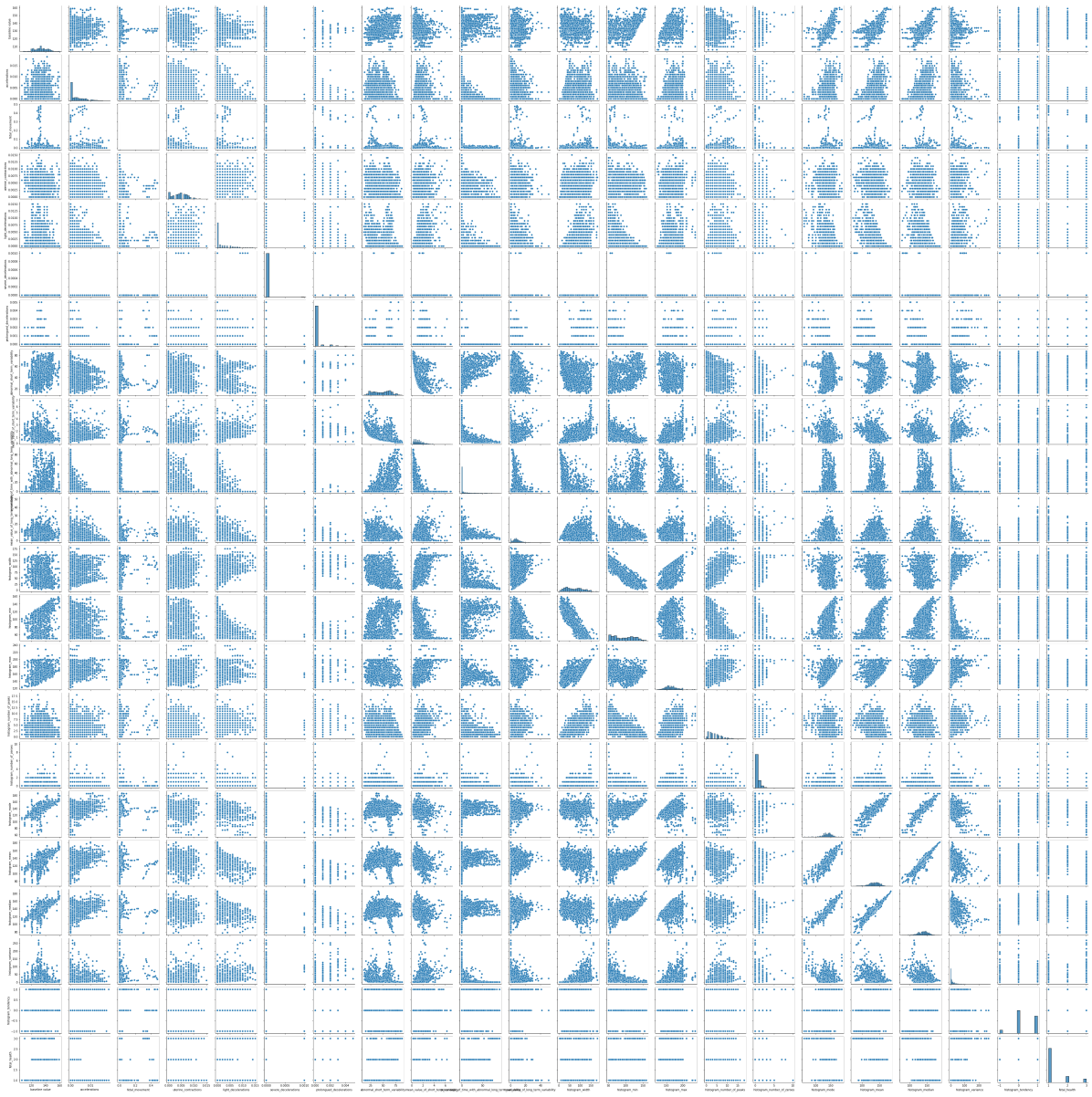
In [20]:

```
for i in fetal_data.columns:
    plt.figure(figsize=(15,6))
    sns.boxplot(fetal_data[i], data = fetal_data)
    plt.xticks(rotation = 90)
    plt.show()
```



In [21]:

```
1 sns.pairplot(fetal_data)
2 plt.show()
```



In [23]:

```
1 fetal_data.corr()
```

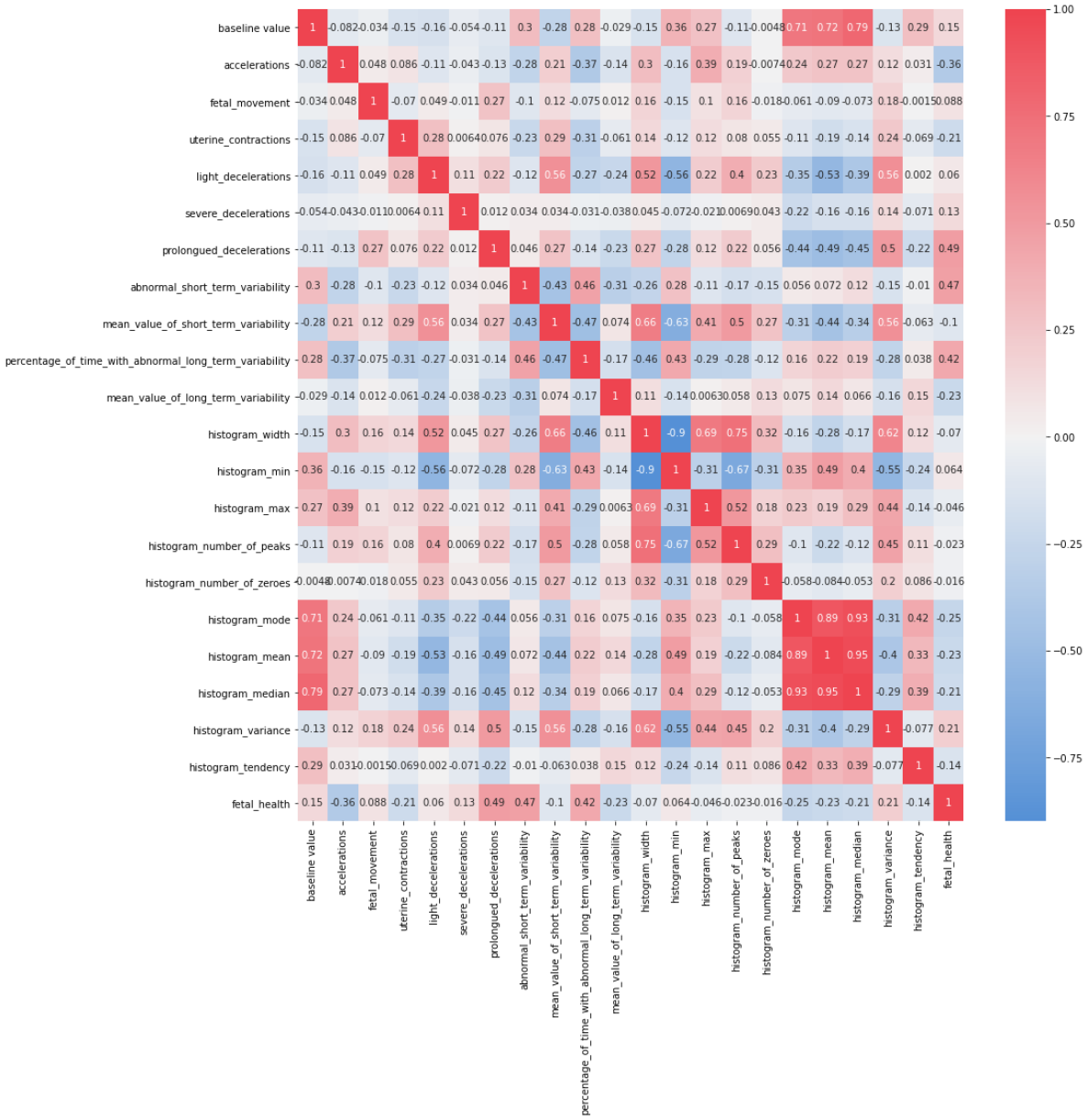
Out[23]:

	baseline value	accelerations	fetal_movement
baseline value	1.000000	-0.081885	-0.033949
accelerations	-0.081885	1.000000	0.048114
fetal_movement	-0.033949	0.048114	1.000000
uterine_contractions	-0.149587	0.086174	-0.069867
light_decelerations	-0.159836	-0.110595	0.048795
severe_decelerations	-0.053706	-0.043237	-0.011022
prolongued_decelerations	-0.105003	-0.128682	0.265802
abnormal_short_term_variability	0.303502	-0.280495	-0.104876
mean_value_of_short_term_variability	-0.278344	0.206762	0.121532
percentage_of_time_with_abnormal_long_term_variability	0.283918	-0.373507	-0.074900
mean_value_of_long_term_variability	-0.028901	-0.141413	0.011749
histogram_width	-0.147150	0.298350	0.162803
histogram_min	0.360129	-0.155306	-0.154297
histogram_max	0.273402	0.392684	0.099703
histogram_number_of_peaks	-0.113242	0.189209	0.164645
histogram_number_of_zeroes	-0.004807	-0.007360	-0.018122
histogram_mode	0.708074	0.243083	-0.061496
histogram_mean	0.722152	0.270266	-0.089938
histogram_median	0.788487	0.272507	-0.072676
histogram_variance	-0.134458	0.124433	0.179115
histogram_tendency	0.294412	0.030670	-0.001459
fetal_health	0.146077	-0.363947	0.088057

22 rows × 22 columns

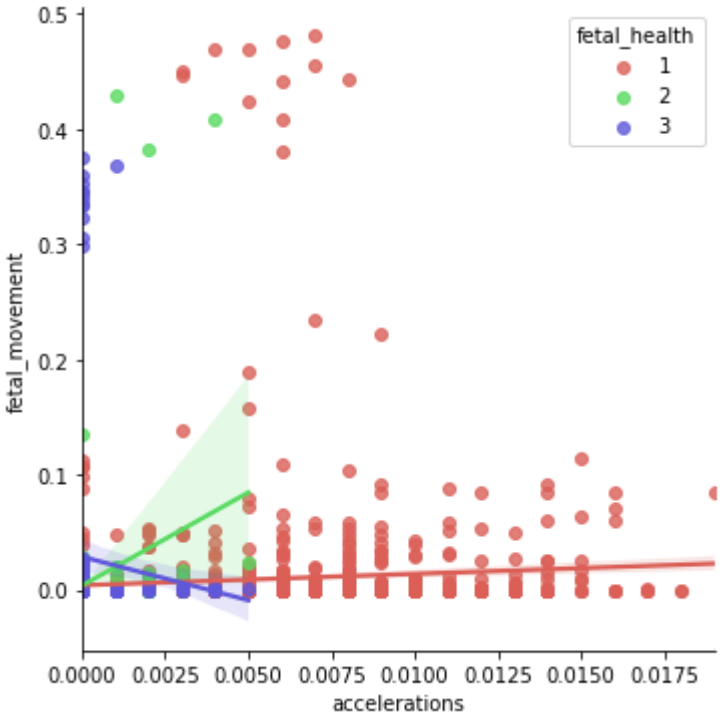
In [25]:

```
plt.figure(figsize=(15,15))
cmap = sns.diverging_palette(250, 10, s=80, l=55, n=9, as_cmap=True)
sns.heatmap(fetal_data.corr(),annot=True, cmap=cmap, center=0)
plt.show()
```



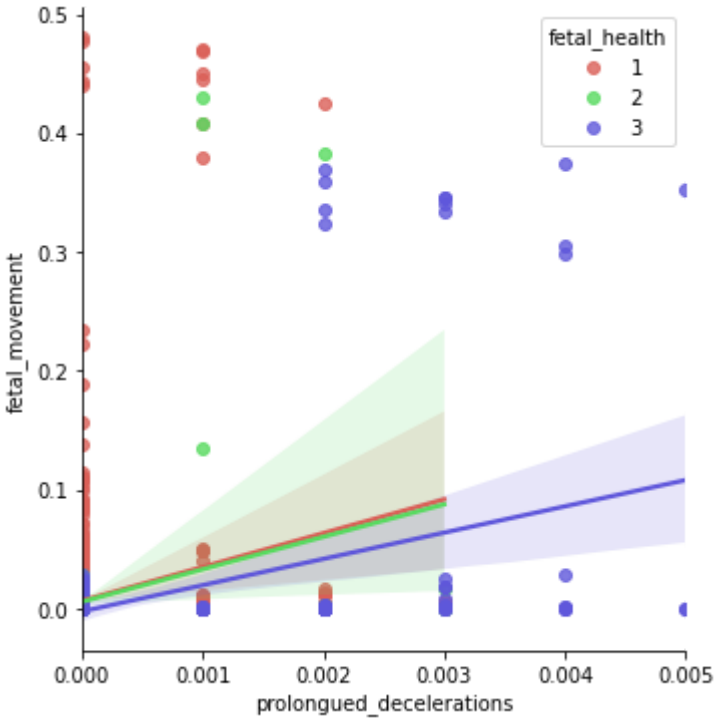
In [27]:

```
sns.lmplot(data = fetal_data,x="accelerations",y="fetal_movement",
           palette='hls', hue="fetal_health",legend_out=False)
plt.show()
```



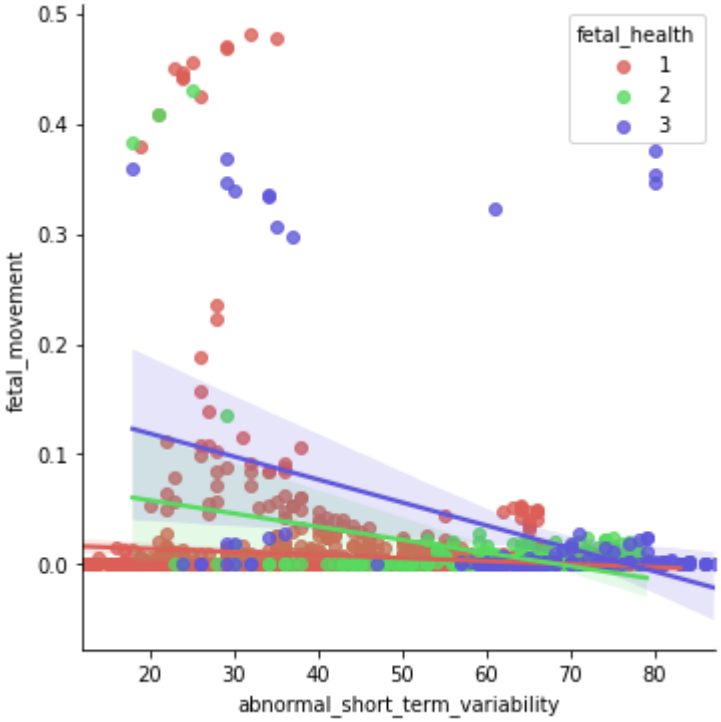
In [29]:

```
sns.lmplot(data = fetal_data,x="prolongued_decelerations",y="fetal_movement",
           palette='hls', hue="fetal_health",legend_out=False)
plt.show()
```



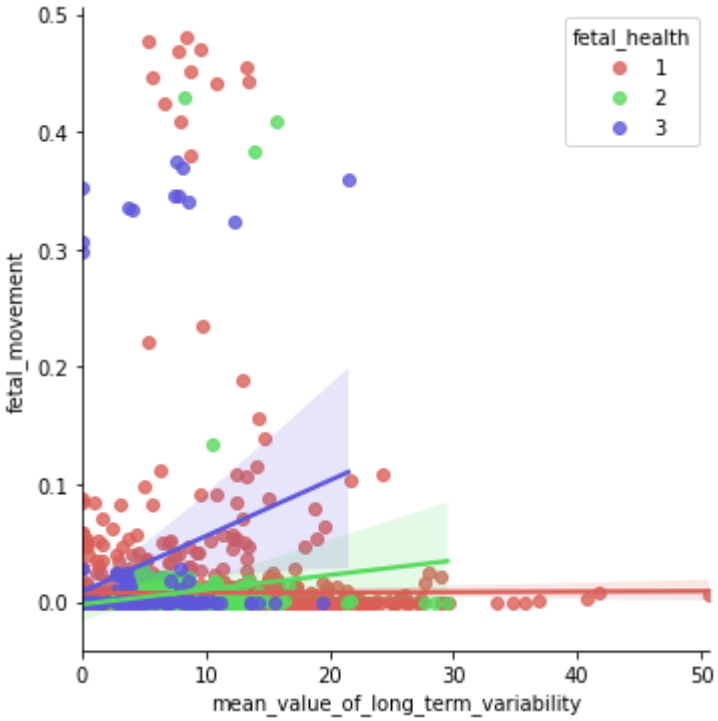
In [30]:

```
sns.lmplot(data = fetal_data,x="abnormal_short_term_variability",y="fetal_movement"
           palette='hls', hue="fetal_health",legend_out=False)
plt.show()
```



In [31]:

```
sns.lmplot(data = fetal_data,x="mean_value_of_long_term_variability",y="fetal_movem
           palette='hls', hue="fetal_health",legend_out=False)
plt.show()
```

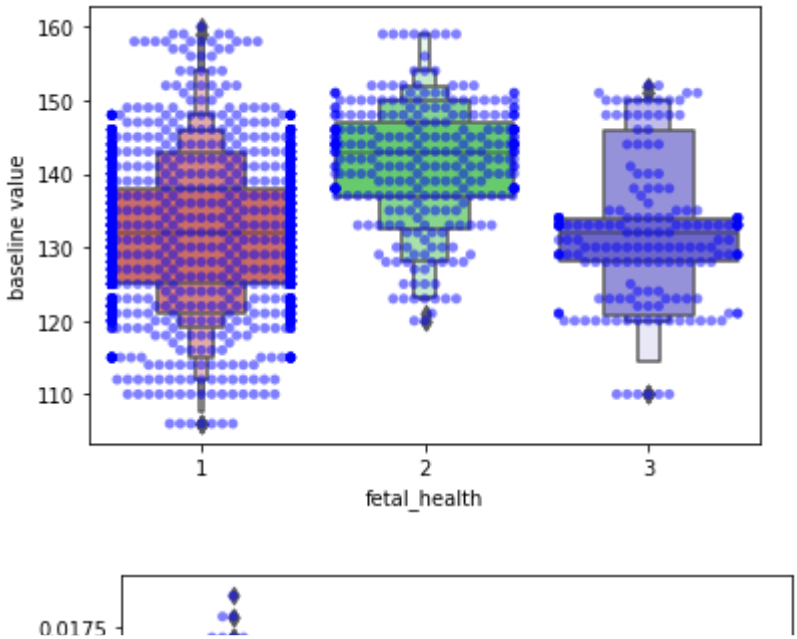


In [32]:

```
fetal_data_cols=['baseline value', 'accelerations', 'fetal_movement',
       'uterine_contractions', 'light_decelerations', 'severe_decelerations',
       'prolongued_decelerations', 'abnormal_short_term_variability',
       'mean_value_of_short_term_variability',
       'percentage_of_time_with_abnormal_long_term_variability',
       'mean_value_of_long_term_variability']
```

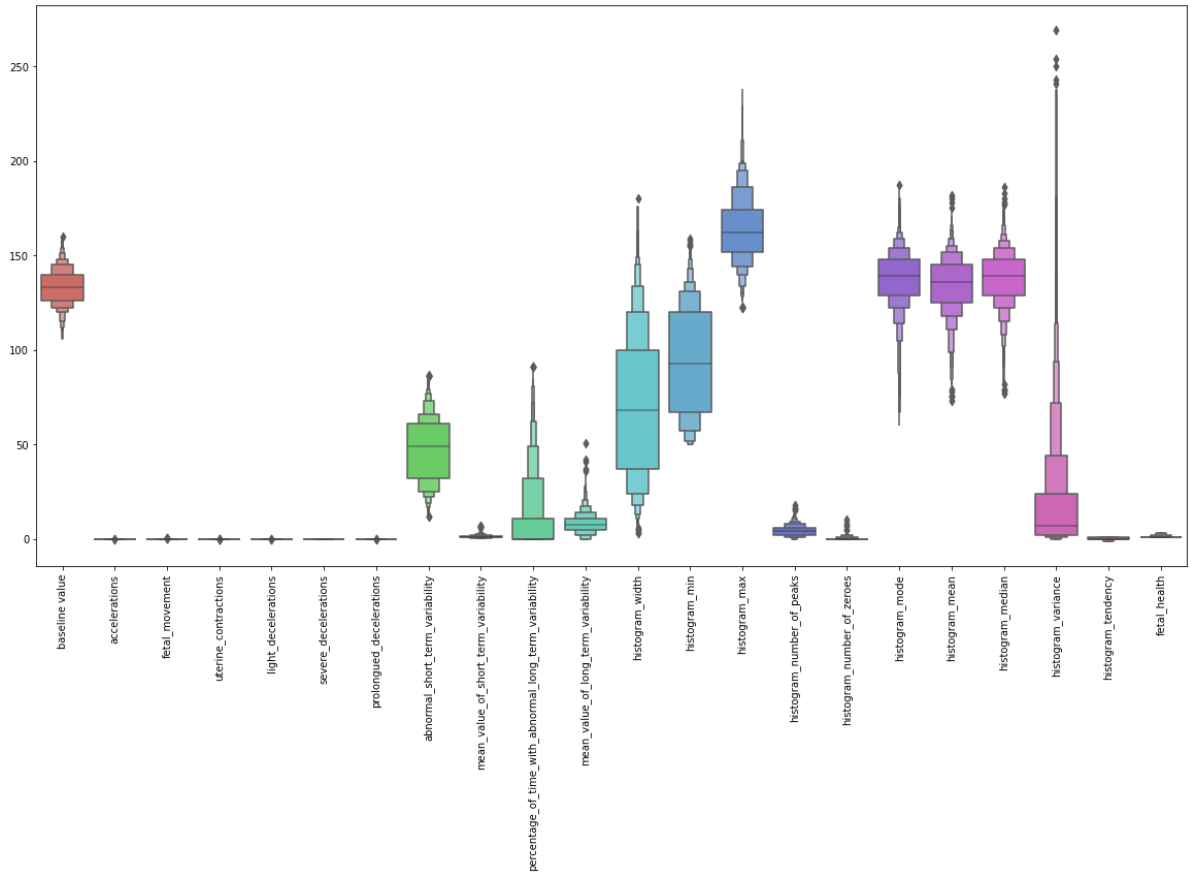
In [33]:

```
for i in fetal_data_cols:
    sns.swarmplot(x=fetal_data["fetal_health"], y=fetal_data[i],
                  color="blue", alpha=0.5 )
    sns.boxenplot(x=fetal_data["fetal_health"], y=fetal_data[i],
                  palette='hls')
    plt.show()
```



In [34]:

```
plt.figure(figsize=(20,10))
sns.boxenplot(data = fetal_data,palette = 'hls')
plt.xticks(rotation=90)
plt.show()
```



In [35]:

```
from sklearn.model_selection import train_test_split
from sklearn import preprocessing
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.svm import LinearSVC
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import cross_val_score
from sklearn.metrics import precision_score, recall_score, confusion_matrix, classi
from sklearn import metrics
from sklearn.metrics import roc_curve, auc, roc_auc_score
np.random.seed(0)
```

In [36]:

```
X=fetal_data.drop(["fetal_health"],axis=1)
y=fetal_data["fetal_health"]
```

In [37]:

```
1 X.shape
```

Out[37]:

```
(2113, 21)
```

In [38]:

```
1 y.shape
```

Out[38]:

```
(2113,)
```

In [39]:

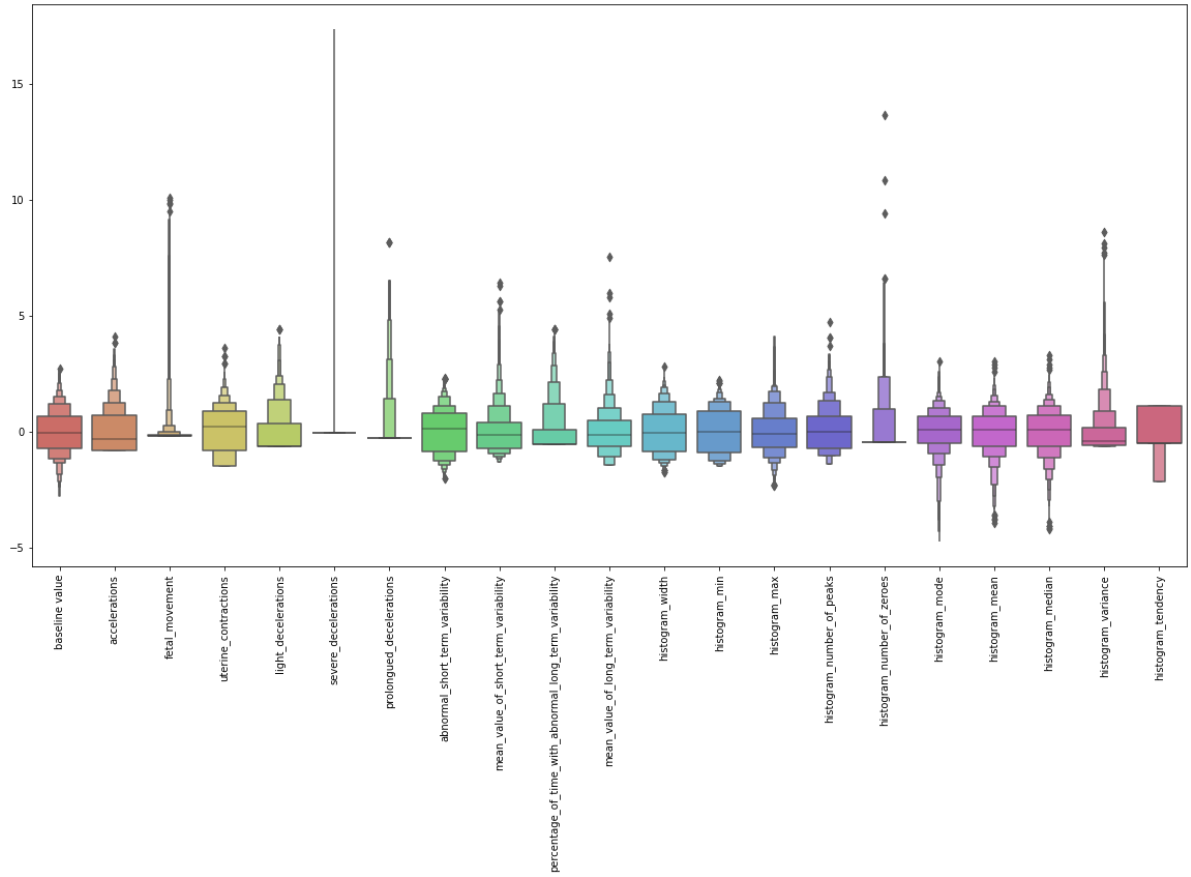
```
col_names = list(X.columns)
s_scaler = preprocessing.StandardScaler()
X_df= s_scaler.fit_transform(X)
X_df = pd.DataFrame(X_df, columns=col_names)
X_df.describe().T
```

Out[39]:

	count	mean	std	m
baseline value	2113.0	-1.143325e-15	1.000237	-2.77625
accelerations	2113.0	6.725440e-18	1.000237	-0.82377
fetal_movement	2113.0	-1.345088e-17	1.000237	-0.20339
uterine_contractions	2113.0	8.070528e-17	1.000237	-1.49224
light_decelerations	2113.0	-8.070528e-17	1.000237	-0.64120
severe_decelerations	2113.0	-6.725440e-18	1.000237	-0.05765
prolongued_decelerations	2113.0	4.035264e-17	1.000237	-0.26964
abnormal_short_term_variability	2113.0	-1.345088e-17	1.000237	-2.03764
mean_value_of_short_term_variability	2113.0	-4.035264e-17	1.000237	-1.28373
percentage_of_time_with_abnormal_long_term_variability	2113.0	2.690176e-17	1.000237	-0.53429
mean_value_of_long_term_variability	2113.0	1.076070e-16	1.000237	-1.45015
histogram_width	2113.0	5.044080e-17	1.000237	-1.73174
histogram_min	2113.0	1.042443e-16	1.000237	-1.47400
histogram_max	2113.0	3.127330e-16	1.000237	-2.34658
histogram_number_of_peaks	2113.0	0.000000e+00	1.000237	-1.38162
histogram_number_of_zeroes	2113.0	7.313916e-17	1.000237	-0.46015
histogram_mode	2113.0	-1.614106e-16	1.000237	-4.72336
histogram_mean	2113.0	3.497229e-16	1.000237	-3.94699
histogram_median	2113.0	3.497229e-16	1.000237	-4.22018
histogram_variance	2113.0	-6.725440e-17	1.000237	-0.65125
histogram_tendency	2113.0	2.690176e-17	1.000237	-2.15819

In [40]:

```
plt.figure(figsize=(20,10))
sns.boxenplot(data = X_df,palette = 'hls')
plt.xticks(rotation=90)
plt.show()
```



In [41]:

```
X_train, X_test, y_train,y_test = train_test_split(X_df,y,
                                                   test_size=0.2,
                                                   random_state=42)
```

In [42]:

```
pipeline_lr=Pipeline([('lr_classifier',LogisticRegression(random_state=42))])
pipeline_dt=Pipeline([ ('dt_classifier',DecisionTreeClassifier(random_state=42))])
pipeline_rf=Pipeline([('rf_classifier',RandomForestClassifier())])
pipeline_svc=Pipeline([('sv_classifier',SVC())])
pipelines = [pipeline_lr, pipeline_dt, pipeline_rf, pipeline_svc]
pipe_dict = {0: 'Logistic Regression', 1: 'Decision Tree',
             2: 'RandomForest', 3: "SVC"}
```

In [43]:

```
for pipe in pipelines:
    pipe.fit(X_train, y_train)
```

In [44]:

```
cv_results_accuracy = []
for i, model in enumerate(pipelines):
    cv_score = cross_val_score(model, X_train,y_train, cv=10 )
    cv_results_accuracy.append(cv_score)
    print("%s: %f " % (pipe_dict[i], cv_score.mean()))
```

```
Logistic Regression: 0.887574
Decision Tree: 0.915385
RandomForest: 0.937278
SVC: 0.904734
```

In [45]:

```
pred_rfc = pipeline_rf.predict(X_test)
```

In [46]:

```
print("Training Accuracy :", pipeline_rf.score(X_train, y_train))
print("Testing Accuracy :", pipeline_rf.score(X_test, y_test))
```

```
Training Accuracy : 0.9994082840236687
Testing Accuracy : 0.966903073286052
```

In [50]:

```
from sklearn.metrics import accuracy_score
```

In [51]:

```
n report for classifier %s:\n%s\n" % (pipeline_rf,
                    metrics.classification_report(y_test, pred_rfc)))
```

```
 Classification report for classifier Pipeline(steps=[('rf_classifier', Ra
ndomForestClassifier())]):
              precision    recall  f1-score   support

           1       0.97      0.99      0.98       341
           2       0.93      0.80      0.86        54
           3       1.00      1.00      1.00        28

    accuracy                           0.97       423
   macro avg       0.97      0.93      0.95       423
weighted avg       0.97      0.97      0.97       423
```

In [52]:

```
matrix = confusion_matrix(y_test, pred_rfc)
print('Confusion matrix : \n',matrix)
```

```
Confusion matrix :
 [[338   3   0]
 [ 11  43   0]
 [  0   0  28]]
```

In [53]:

```
plt.subplots(figsize=(12,8))
sns.heatmap(matrix/np.sum(matrix), cmap=cmap,annot = True,
            annot_kws = {'size':15})
plt.show()
```